Integrating Handhelds with ACEs: "Leashing" Devices Using a Services-Based Architecture

Jeffrey Eschbach
Motorola Labs
eschbach@labs.mot.com

Nitya Narasimhan
Motorola Labs
nitya@labs.mot.com

Kabe Vander Baan
Motorola Labs
kabev@labs.mot.com

Abstract

Both ACEs and handheld devices are becoming ubiquitous in scientific and research communities; however, the two have not been effectively integrated. We propose a mechanism we term "leashing" that enables a device to share a node's current virtual space while ensuring that the device's resources are not overwhelmed. We implement the leashing mechanism in an extensible manner by employing a services-based architecture.

Keywords: *handheld, leashing, services, collaborative environments.*

1. Introduction

Advanced Collaborative Environments (ACEs) [1] provide rich, interactive audio/video content to foster *presence* between geographically remote scientists. An ACE consists of a collection of *virtual* spaces, each spanning multiple *physical* spaces. A physical space consists of components that traverse virtual spaces as a single unit, while the virtual spaces provide a setting for sharing content across the physical spaces. For example, virtual venues in Access Grid (AG) nodes [2,3] advertise multicast addresses to share audio/video content across the associated physical spaces. For convenience, and in keeping with standard terminology, we use the term "venue" to refer to a virtual space and the term "node" to refer to a physical space.

ACEs typically employ fixed infrastructures consisting of tethered, resource-rich machines operating over wired networks. Wireless, handheld devices can extend the utility of such ACEs with support for mobile operation, enhanced maneuverability, and an ability to personalize a user's experience. Most current approaches to integrating handhelds into ACEs view the handheld as a stand-alone device [4] that can replicate node functionality; however, such approaches have two basic limitations.

First, many users may prefer to associate the handheld device with an *existing* physical space rather than have the device navigate virtual spaces in an independent manner. Second, the resources of such devices are easily overwhelmed (e.g., bandwidth, processing power, display area) by the amount of traffic present in broadband multimedia environments such as ACEs.

To overcome these limitations, we propose to integrate handhelds into ACEs using a mechanism that we term "leashing". Leashing performs multiple functions: it *associates* a device with a particular node, *maintains* the device in the context of the node's virtual space as the node traverses different venues, and *regulates* the interaction of the device with its environment in a manner that suits the device's resource constraints. While a variety of devices could exploit the leashing mechanism, it is particularly applicable to wireless handhelds.

2. The Leashing Mechanism

We facilitate this leashing approach by deploying a "dock" within a node which devices register with to associate themselves with that physical space. Because the leashing dock is an intrinsic component of the node, it moves with the node as it travels through virtual spaces. Consequently, all leashed devices are automatically "towed" by the dock through these virtual spaces. All leashed devices will therefore always share the same venue as the node without requiring further configuration.

To accommodate the resource limitations of a leashed device, the dock will regulate the traffic shared between the device and the current venue. A "basic" leashing dock could function as a simple filter, forwarding to all devices the same *subset* of the traffic available at the venue. If the bandwidth is a limiting factor (e.g., a wireless interface between dock and handhelds), the dock could establish a dedicated multicast address for forwarding content to all leashed devices. The devices would use this address to receive the selected subset of venue traffic.

An "advanced" dock could provide additional features to better accommodate specific limitations per device. If the leashed devices' hardware capabilities were a limiting factor (e.g., processing power, display size), the dock could support unicast connections; handhelds could subsequently register to receive individual unicast feeds that are tailored to accommodate their specific constraints. As an example, a device could inform the dock of the maximum number of video streams that it can send and receive at a time; the dock then monitors traffic and forwards only the appropriate number of streams to the device (see Figure 1). In all scenarios, to support user transparency, the handheld-to-dock interface remains unchanged even as the node moves across different venues (see Figure 2).

Options for traffic selection can also vary widely. While a simple implementation could forward a pre-selected video feed (e.g., the main camera feed associated with the node) to all leashed handhelds, other implementations could enable selection of feeds through a menu interface [4], either by the node operator or handheld user. Alternatively, more complex algorithms could be employed to select the subset of feeds (e.g., based on audio and video activity levels) or even to modify the traffic (e.g., by reducing the frame rate) to accommodate processing and/or bandwidth constraints.

Additional issues surrounding the implementation of leashing include the following items:

- Identification of use cases (e.g., remote monitoring of a node of interest, personalization of user content).
- Methods to determine and accommodate leashed device's capabilities.
- Adaptation to dynamic resource constraints and to the addition/removal of leashed devices.
- Leashing applicability to devices besides handhelds.

3. Services-Based Architecture for Leashing

The heterogeneity of handheld devices and the diversity of ACE usage scenarios will fuel the development of a variety of docking implementations. Users would therefore be required to reconfigure their handhelds manually when they leash to different nodes. This reduces the scalability, transparency and utility of the leashing model.

This problem can be addressed by deploying a services-based architecture within the nodes. The architecture would view leashing implementations as "services" with well-defined interfaces and attributes that can be extended, modified, added or removed dynamically. Like typical services-based models [5,6], a service architecture for leashing must expose mechanisms for *service definition* (i.e., by the provider), *service advertisement* (i.e., by the service broker) and *service discovery* (i.e., by clients of the service). Instead of maintaining a separate configuration for each docking implementation, a handheld could use these mechanisms to access any leashing dock (implemented as a service) deployed within a node.

With a services-based architecture for leashing, key issues emerge that require further investigation:

- Effective standardization of service attributes and interfaces to automate configuration.
- Automatic detection of brokers (e.g., as devices move from one physical space to another).
- Value of deploying multiple brokers (e.g., for fault tolerance, availability and load balancing).
- Association of devices with multiple brokers, within the same node and/or across different nodes.
- Ability for service brokers and devices to identify each other.
- Define the "core" (mandatory) services for a particular ACE.

There is also future work in this space beyond leashing. The architecture's extensibility can be leveraged to define additional services for handhelds. Further, the services-based architecture can be extended beyond the node to the venues; a service broker and services could be available at each venue, and/or services could be shared between nodes residing in the same venue.

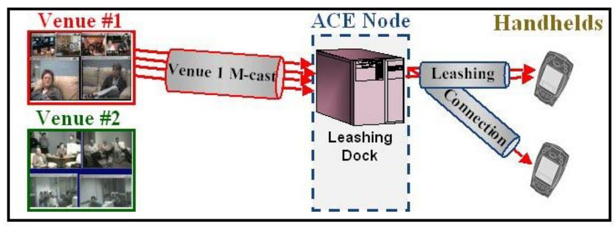

Figure 1: Handhelds connect to the leashing dock; ACE node receives feeds (red arrows) for venue #1; dock forwards a subset of feeds to the handhelds.

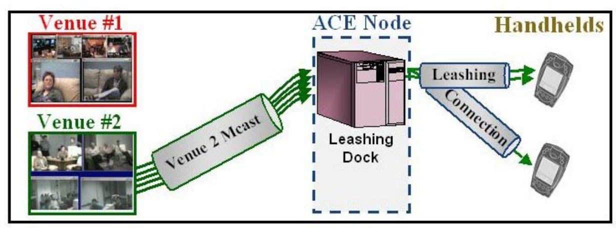

Figure 2: ACE node changes venues and receives new feeds (green arrows); leashing connection remains unchanged; dock forwards a subset of the new feeds.

References

[1] http://www-fp.mcs.anl.gov/fl/wace/, May 2002
[2] http://www.accessgrid.org, May 2002
[3] http://www-fp.mcs.anl.gov/fl/accessgrid/downloads/AG-market-2002.pdf , May 2002
[4] M.Thorson, J. Leigh, G. Maajid, K. Park, At. Nayak, P. Salva, S. Berry "AccessGrid-to-Go: Providing AccessGrid access on Personal Digital Assistants", http://www-fp.mcs.anl.gov/fl/accessgrid/ag-retreat-2002/abstracts/leigh-ag2g-finalpaper.pdf, March 2002
[5] J. Veizades, E. Guttman, C. Perkins and S. Kaplan, "Service Location Protocol", RFC 2165, June 1997.
[6] K. Arnold, B. O'Sullivan, R. Scheifler, J. Waldo, and A. Wollrath, "The Jini Specification", Addison-Wesley, 1999